



Prioritätsbescheinigung über die Einreichung einer Patentanmeldung

Aktenzeichen: 103 08 815.6

Anmeldetag: 27. Februar 2003

Anmelder/Inhaber: Siemens Aktiengesellschaft, München/DE

Bezeichnung: Verfahren zur Erzeugung und Visualisierung einer aufgabenorientierten Schrittdarstellung

IPC: G 06 F 3/14

Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der ursprünglichen Unterlagen dieser Patentanmeldung.

München, den 2. Juli 2003
Deutsches Patent- und Markenamt
Der Präsident
Im Auftrag

Jerofsky

Beschreibung

Verfahren zur Erzeugung und Visualisierung einer aufgabenorientierten Schrittdarstellung

5

Die Erfindung betrifft ein Verfahren zur Erzeugung und Visualisierung einer aufgabenorientierten Schrittdarstellung von mindestens einem Teileprogramm bei Werkzeug- oder Produktionsmaschinen.

10

Bei Werkzeug- oder Produktionsmaschinen, wobei unter Produktionsmaschinen auch Roboter zu verstehen sind, werden innerhalb eines Produktionsprozesses zur Bearbeitung eines Werkstückes, z.B. Achsen und/oder Spindeln bewegt. Die Achsen und/oder Spindeln, die hierbei eine Relativbewegung zwischen einem Werkzeug und einem Werkstück beschreiben, werden als sogenannte Bearbeitungseinheiten bezeichnet. Die Achsen und/oder Spindeln werden für Bearbeitungen einem sogenannten Kanal zugeordnet. Die Bewegungsaufträge an die Bearbeitungseinheit werden in dem Kanal in Form eines Teileprogramms vorgegeben und beschrieben. Das Teileprogramm wird innerhalb einer numerischen Steuerung der Maschine einem Interpreter übergeben, der das Teileprogramm in einen entsprechenden Maschinencode umsetzt.

25

Zur Erhöhung der Produktivität verfügen Werkzeug- oder Produktionsmaschinen oft über mehrere Bearbeitungseinheiten, die simultan bewegt werden können. Solche Maschinen werden als mehrkanalige Maschinen bezeichnet. Bei diesen werden zur simultanen Bearbeitung mehrere Relativbewegungen zwischen Werkstück/en und Werkzeug/en in eigenständigen Teileprogrammen definiert, die simultan von mehreren Kanälen interpretiert und abgefahren werden. Jedem Kanal ist dabei ein eigenes Teileprogramm zugeordnet.

35

Ein Teileprogramm besteht dabei in der Regel aus einem Standard ASCII-Source-Code gemäß DIN 66025/ISO, sowie gegebenen-

falls zusätzliche hersteller- bzw. maschinenspezifischer Erweiterungen. Die Teileprogramme werden üblicherweise in ASCII-Darstellung mit einem Editor insbesondere einem Texteditor angezeigt und bearbeitet. Daneben gibt es Programmiersysteme zur Programmierung der Maschine bzw. Erstellung von
5 Teileprogrammen mit deren Hilfe diese aufgabenorientiert innerhalb einer sogenannten aufgabenorientierten Schrittdarstellung erstellt und dem Bediener visualisiert werden. Eine Schrittdarstellung erlaubt durch die strukturierte Ansicht
10 und Bearbeitungsmöglichkeit der einzelnen Teileprogramme eine leichtere Programmierung und Bedienung der Maschine. Bislang konnten jedoch nur Programme in einer Schrittdarstellung bzw. in Schritten dargestellt werden, wenn sie auch von einem schrittdarstellungsunterstützenden Programmiersystem erzeugt
15 worden sind. Vorhandene, bereits bestehende Teileprogramme konnten nicht nachträglich mittels einer Schrittdarstellung bearbeitet werden. Insbesondere an der Maschine vor Ort wurden vom Bediener die Teileprogramme nur mit einem reinen Texteditor visualisiert und bearbeitet.

20 Der Erfindung liegt daher die Aufgabe zugrunde ein einfaches Verfahren zur Erzeugung und Visualisierung einer aufgabenorientierten Schrittdarstellung von mindestens einem Teileprogramm bei Werkzeug- oder Produktionsmaschinen zu schaffen.

25 Die Aufgabe wird für das erfindungsgemäße Verfahren dadurch gelöst, dass das Teileprogramm von einem Syntaxanalysator nach Schlüsselbegriffen durchsucht wird, dass anhand der solchermaßen gefundenen Schlüsselbegriffe eine aufgabenorientierte Schrittdarstellung des Teileprogramms erzeugt und einem Bediener visualisiert wird.
30

35 Eine erste vorteilhafte Ausbildung des erfindungsgemäßen Verfahrens ist dadurch gekennzeichnet, dass als Schlüsselbegriffe fest definierte Anweisungen, insbesondere Synchronisationsanweisungen und/oder Anwenderzyklusanweisungen, bzw. Syntaxteile der Anweisungen verwendet werden. Durch die Verwen-

dung von fest definierten Anweisungen bzw. Syntaxteile der Anweisungen als Schlüsselbegriffe wird eine besonders zuverlässige und sichere Erzeugung der Schrittdarstellung des Teileprogramms sichergestellt.

5

Eine weitere vorteilhafte Ausbildung der Erfindung ist dadurch gekennzeichnet, dass die Schlüsselbegriffe sowie deren Zuordnung zu den jeweiligen Schritten in einer vom Anwender veränderbaren Konfigurationsdatei abgelegt sind und vom Syntaxanalysator eingelesen werden. Insbesondere auf die häufig von den Maschinenherstellern verwendeten programmiersprachlichen Erweiterungen im Hinblick auf den Standard ASCII-Source-Code nach DIN 66025/ISO erlaubt eine konfigurierbare Konfigurationsdatei eine flexible auf den jeweiligen Hersteller und die jeweilige Maschine zugeschnittene Auswahl der Schlüsselbegriffe.

10

15

20

Eine weitere vorteilhafte Ausbildung der Erfindung ist dadurch gekennzeichnet, dass die jeweilige Bezeichnung der Schritte in der Konfigurationsdatei abgelegt ist. Hierdurch ist eine einfache Änderungsmöglichkeit der Bezeichnung der Schritte sichergestellt.

25

30

Eine weitere vorteilhafte Ausbildung der Erfindung ist dadurch gekennzeichnet, dass die Visualisierung mittels eines Editors durchgeführt wird. Die Visualisierung mittels eines Editors ist besonders vorteilhaft, da nach der Visualisierung mit Hilfe des Editors der Bediener auch gleich in die Lage versetzt wird, die erzeugte Schrittdarstellung bzw. die ursprüngliche ASCII-Source-Code Darstellung anschließend direkt zu editieren.

35

In diesem Zusammenhang erweist es sich als vorteilhaft, dass durch den Editor mindestens zwei Teileprogramme gleichzeitig in Schritt- und/oder ASCII-Source-Code Darstellung visualisiert werden. Hierdurch können gegebenenfalls auch mehrere Teileprogramme gleichzeitig mit dem Editor bearbeitet werden.

Ein weiteres Ausführungsbeispiel der Erfindung ist dadurch gekennzeichnet, dass die Schrittdarstellung als normierte und/oder synchronisierte Schrittdarstellung visualisiert wird. Eine Darstellung der Schritte in normierte und/oder
5 synchronisierter Form erlaubt eine besonders übersichtliche Darstellung der einzelnen Schritte bzw. des Herstellungsprozesses.

Eine weitere vorteilhafte Ausführung der Erfindung ist dadurch gekennzeichnet, dass die einzelnen durchgeführten Aufgaben in der Schrittdarstellung für jeden Schritt durch
10 schrittspezifische Symbole graphisch dargestellt werden. Eine graphische Darstellung der Schritte mittels Symbole erlaubt eine besonders einfache und übersichtliche Visualisierung der
15 Schritte sowie eine besonders schnelle Erfassung des Schrittyps durch den Anwender.

Eine weitere vorteilhafte Ausführung der Erfindung ist dadurch gekennzeichnet, dass mit Hilfe von Strukturierungsanweisungen im Teileprogramm und/oder mit Hilfe eines Zuweisungsparameters in der Konfigurationsdatei (8), mehrere
20 Schritte zu einem übergeordneten Schritt oder einer hierarchischen Ebene zusammengefasst werden. Hierdurch wird der Aufbau von hierarchischen Schrittstrukturen ermöglicht.

Eine weitere vorteilhafte Ausführung der Erfindung ist dadurch gekennzeichnet, dass mit Hilfe von Strukturierungsanweisungen im Teileprogramm, Schrittbezeichnungen, Symbole und/oder hierarchische Ebenen unabhängig von der Konfigurationsdatei auch direkt im Teileprogramm hinterlegt werden. So
30 mit können auch unabhängig von der Konfigurationsdatei individuell für das jeweilige Teileprogramm zugeschnittene Schrittbezeichnungen und Symbole definiert werden.

Eine weitere vorteilhafte Ausführung der Erfindung ist dadurch gekennzeichnet, dass mit Hilfe einer Definitionsdatei eine Hervorhebung von definierten Anweisungen und/oder
35

Schlüsselbegriffen auf einem Editor visualisiert wird. Hierdurch können dem Anwender z.B. herstellerspezifische Anweisungen nach DIN-Code und Kommentare unterschiedlich dargestellt werden. Dem Anwender wird hierdurch eine bessere Lesbarkeit und ein schnelleres Erfassen der Anweisungen und/oder Schlüsselbegriffen ermöglicht.

Ein Ausführungsbeispiel der Erfindung ist in der Zeichnung dargestellt und wird im folgenden näher erläutert. Dabei zeigen:

- FIG 1 ein Teileprogramm,
- FIG 2 ein Übersichtschaltbild,
- FIG 3 eine Schrittdarstellung des Teileprogramms und
- 15 FIG 4 eine synchronisierte Schrittdarstellung

In FIG 1 ist in Form einer blockförmigen schematischen Darstellung ein Teileprogramm 1 dargestellt. Das Teileprogramm 1 setzt sich aus einer Vielzahl von Anweisungen zusammen, von denen der Übersichtlichkeit halber in Form von schematischen Blöcken nur die Anweisungen 2a, 2b, 2c, 2d, 2e und 2f dargestellt sind. Die einzelnen Anweisungen 2a, 2b, 2c, 2d, 2e und 2f sind dabei in Form eines ASCII-Source-Codes gemäß DIN 66025/ISO geschrieben. Zusätzlich können noch weitere her-

25 steller- und/oder maschinenspezifische Anweisungen in ASCII-Source-Code vorhanden sein. Als Anweisung 2b steht im Ausführungsbeispiel z.B.

N50 G0 X100 Y100

30

was z.B. bedeutet, dass die x-Achse und die y-Achse der Maschine im Eilgang die Position 100,100 mit dem Werkzeug anfahren sollen. Die Anweisung 2c lautet z.B.

35 N60 X150

was bedeutet, dass die x-Achse mit Vorschub zur Position 150 gefahren wird. Da ein Teileprogramm sehr viele Anweisungen enthalten kann und oft auch bei mehrkanaligen Maschinen es notwendig ist, mehrere Teileprogramme untereinander zu koordinieren, ist es sinnvoll und notwendig, dem Anwender eine übersichtlichere Darstellung des Teileprogramms in Form einer sogenannten aufgabenorientierten Schrittdarstellung zur Verfügung zu stellen. Hier nun setzt das erfindungsgemäße Verfahren an.

10

In FIG 2 ist das erfindungsgemäße Verfahren in Form eines Blockschaltbildes dargestellt. Das Teileprogramm 1 wird zunächst von einem Syntaxanalysator 7 nach Schlüsselbegriffen durchsucht. Anhand der solchermaßen gefundenen Schlüsselbegriffe wird eine aufgabenorientierte Schrittdarstellung des Teileprogramms 1 erzeugt und einem Bediener mit Hilfe eines Editors 9 visualisiert. Eine Konfigurationsdatei 8, welche vom Syntaxanalysator 7 eingelesen wird, enthält Informationen welcher Schritt dem jeweiligen Schlüsselbegriff zugeordnet wird. Ein Schlüsselbegriff kann dabei Teil einer Anweisung oder auch eine komplette Anweisung des Teileprogramms 1 sein. Weiterhin enthält die Konfigurationsdatei 8 optional Informationen, welcher Typ von Symbol, dem jeweiligen Schlüsselbegriff bzw. dem jeweiligen Schritt zugeordnet wird. Die Konfigurationsdatei 8 kann vom Anwender entsprechend seinen Anforderungen und Bedürfnissen verändert bzw. konfiguriert werden.

25

In FIG 3 ist in Form einer schematisierten Darstellung eine aufgabenorientierte Schrittdarstellung 5 beispielhaft dargestellt, wie sie z.B. auch einem Bediener der Maschine auf z.B. einem Monitor visualisiert wird. In FIG 3 sind beispielhaft die vier Schritte 6a, 6b, 6c und 6d dargestellt. Der Schritt 6a ist dabei mit „Gewinde schneiden“ bezeichnet, der Schritt 6b ist mit „freier DIN-Code“ bezeichnet, der Schritt 6c ist mit „Oberflächen glätten“ bezeichnet und der Schritt 6d ist mit „WAITM(1,1,2)“ bezeichnet. Zusätzlich wird der Schritt 6a durch ein Symbol 4a graphisch dargestellt. Ent-

35

sprechend wird auch der Schritt 6b, 6c und 6d durch die Symbole 4b, 4c bzw. 4d graphisch dargestellt. In Bezug auf die gezeigte Schrittdarstellung sind die Symbole hierbei optional zu sehen.

5

Zu Beginn des Verfahrens sucht der Syntaxanalysator 7 gemäß FIG 2 das Teileprogramm 1 nach Schlüsselbegriffen ab. Im Ausführungsbeispiel sind in der Konfigurationsdatei 8 die Schlüsselbegriffe CYC90, CYC70 und WAITM definiert, wobei dem Schlüsselbegriff CYC90 der Schritt 6a und das Symbol 4a zugeordnet wurde. Entsprechend wurde dem Schlüsselbegriff CYC70 der Schritt 6c und das Symbol 4c zugeordnet und dem Schlüsselbegriff WAITM der Schritt 6d und das Symbol 4d zugeordnet

10

In dem Teileprogramm 1 gemäß FIG 1 lautet die Anweisung 2a [N40 CYC90(...)] die Anweisung 2e [N80 CYC70(...)] und die Anweisung 2f [N100 WAITM(1,1,2)]. Als Teil der Anweisungen werden dabei häufig in Klammern zusätzliche Attribute übergeben, die im Ausführungsbeispiel in Form von Pünktchen angegeben sind.

20

Stößt nun der Syntaxanalysator 7 gemäß FIG 2 auf den Schlüsselbegriff CYC90 in der Anweisung 2a, so erzeugt er in der Schrittdarstellung 5 den Schritt 6a, mit der Bezeichnung „Gewindeschneiden“, sowie das zugehörige Symbol 4a. Anschließend durchsucht der Syntaxanalysator 7 weiter das Teileprogramm 1 und zwar solange bis er wieder einen weiteren Schlüsselbegriff findet. Die Anweisungen 2b, 2c, 2d enthalten im Ausführungsbeispiel keinen Schlüsselbegriff und liegen zwischen den beiden Schlüsselbegriffen CYC90 und CYC70. Folglich werden diese Anweisungen vom Syntaxanalysator 7 zu einem Schritt 6b mit der Bezeichnung „freier DIN-Code“ zusammengefasst und das zugehörige Symbol 4b erzeugt. Sowohl die Bezeichnung als auch das Symbol sind dabei in der Konfigurationsdatei 8 definiert.

In dem Ausführungsbeispiel findet der Syntaxanalysator in der Anweisung 2e [N80 CYC70(...)] den nächsten Schlüsselbegriff CYC70. Da im Ausführungsbeispiel der Schlüsselbegriff CYC70

25

30

35

in der Konfigurationsdatei 8 mit dem Schritt 6c, der die Bezeichnung „Oberflächen glätten“ hat, verbunden ist, erzeugt der Syntaxanalysator 7 in der Schrittdarstellung 5, den Schritt 6c mit der Bezeichnung „Oberflächenglätten“ sowie das zugehörige Symbol 4c. Anschließend untersucht der Syntaxanalysator 7 erneut das Teileprogramm 1 gemäß FIG 1 solange bis er den nächsten Schlüsselbegriff findet, der im Ausführungsbeispiel durch den Schlüsselbegriff WAITM gegeben ist. Gemäß der Zuordnung in der Konfigurationsdatei 8 wird ein entsprechender Schritt 6d mit der Bezeichnung WAITM(1,1,2) erzeugt, sowie ein dazugehöriges Symbol. In diesem Fall liegt ein Spezialfall vor, da der Schlüsselbegriff mit der Bezeichnung des Schrittes 6d weitgehend identisch ist.

Auf diese Art und Weise werden zusammengehörige Anweisungen jeweils zu einem Schritt zusammengefasst. Die Anweisung, die den Schritt 6a (Gewinde scheiden) in FIG 3 umfasst, ist in FIG 1 gestrichelt eingerahmt und mit dem Bezugszeichen 3a versehen. Die Anweisungen, die zum Schritt 6b (freier DIN-Code) gehören, sind in FIG 1 mit dem Bezugszeichen 3b versehen. Die Anweisung, die zum Schritt 6c (Oberflächen glätten) gehört, ist in FIG 1 mit dem Bezugszeichen 3c versehen. Die zum Schritt 6d zugehörige Anweisung ist in FIG 1 ebenfalls gestrichelt umrahmt und mit dem Bezugszeichen 3d versehen. Mit Hilfe des erfindungsgemäßen Verfahrens wird somit aus dem unübersichtlichen Teileprogramm 1 eine strukturierte für den Bediener übersichtliche Schnittdarstellung 5 erzeugt und mit dem Editor 9 visualisiert.

An dieser Stelle sei bemerkt, dass selbstverständlich neben den im Ausführungsbeispiel verwendeten Schlüsselbegriffen, auch andere Schlüsselbegriffe mit zum Teil auch unterschiedlichen Wirkungen auf den Syntaxanalysator 7 verwendet werden können. So sind zum Beispiel auch spezielle Schlüsselbegriffe möglich, die z.B. den Syntaxanalysator 7 veranlassen, bis zum Auffinden des nächsten Schlüsselbegriffes, alle dazwischenliegende Anweisungen zu einem Schritt zusammenzufassen und

den Schritt entsprechend dem speziellen Schlüsselbegriff zu bezeichnen bzw. ein entsprechendes Symbol zu erzeugen.

Wie schon in der Einleitung der Beschreibung erwähnt, wird bei mehrkanaligen Maschinen jedem Kanal ein eigenes Teileprogramm zugeordnet. Mit Hilfe eines entsprechenden Anwahlmenüs kann der Bediener bei solchen Maschinen mehrere Teileprogramme vom Syntaxanalysator in eine Schrittdarstellung umwandeln und gegebenenfalls gleichzeitig auf dem Editor 9 visualisieren lassen. Dadurch, dass z.B. zwei Teileprogramme mittels des Editors 9 gleichzeitig visualisiert werden können, lassen sich die beiden Programme hinsichtlich ihres Ablaufs miteinander vergleichen. Der Editor 9 bietet hierzu zur Unterstützung eine sogenannte normierte Schrittdarstellung und eine sogenannte synchronisierte Schrittdarstellung zur Visualisierung an. Bei der normierten Schrittdarstellung werden die einzelnen Schritte hintereinander auf dem Monitor dargestellt.

In der synchronisierten Schrittdarstellung, die in FIG 4 dargestellt ist, werden die beiden Schrittdarstellungen 5 und 10 von zwei unterschiedlichen Teileprogrammen nebeneinander jeweils so dargestellt, dass spezielle Schritte wie z.B. Schritte, die eine Synchronisation zwischen den beiden Teileprogrammen anzeigen und/oder Koordinierungsanweisungen auf einer gemeinsamen Zeilenebene dargestellt werden.

Ein solcher spezieller Synchronisationsschritt stellt z.B. der Schritt 6d [WAITM(1,1,2)] bzw. 11d [WAITM(1,1,2)] in FIG 4 dar. Im Ausführungsbeispiel gemäß FIG 4 ist neben der Schrittdarstellung 5, die Schrittdarstellung 10 eines zweiten Teileprogramms dargestellt. Im Ausführungsbeispiel wird die Ausführung des zweiten Teileprogramms mit Erreichen des Schritts 11d [WAITM(1,1,2)] zur Synchronisation der beiden Teilprogramme so lange angehalten bis das in der Schrittdarstellung 5 dargestellte Teileprogramm 1 den Schritt 6d erreicht hat. Anschließend werden die beiden Teilprogramme ge-

meinsam fortgesetzt. Besitzt dabei ein Teileprogramm in der Schrittdarstellung, innerhalb von zwei solchen speziellen Schritten zur Synchronisation und/oder Koordinierung von Teileprogrammen, mehr Schritte als das andere Teileprogramm, dann wird in der Schrittdarstellung das Teileprogramm, das weniger Schritte aufweist, in der Darstellung mit entsprechenden Leerschritten bzw. im Ausführungsbeispiel mit einem Leerschritt lld aufgefüllt, wobei in FIG 4, der Übersichtlichkeit halber, der vor dem „Gewinde scheiden“ bzw. „Bohren“ kommende Synchronisations- bzw. Koordinierungsschritt nicht mehr dargestellt ist.

Als Schlüsselbegriffe eignen sich hierbei besonders, die schon angesprochenen Synchronisationsanweisungen oder Syntaxteile von ihnen, wie z.B. WAITM und/oder auch Anwenderzyklusanweisungen, d.h. Sprünge in Unterprogramme wie z.B. der Anweisungsteil CYC sowie spezielle Koordinierungsanweisungsteile z.B. GET/RELEASE von gemeinsam genutzten Ressourcen (z.B. Achsen unter den Kanälen tauschen). Die Synchronisationsanweisung WAITM stellt z.B. eine Anweisung zur Programmkoordination von Teileprogrammen dar, die den Programmablauf der Teileprogramme in der Abarbeitung beeinflusst. Mit Hilfe der Konfigurationsdatei, welche vom Anwender frei konfiguriert werden kann, können im Prinzip beliebige Syntaxfolgen bzw. Teile von Anweisungen oder ganze Anweisungen in die Konfigurationsdatei 8 geschrieben werden und dort mit den ebenfalls frei konfigurierbaren Namensbezeichnungen der zugehörigen Schritte und den zugehörigen Symbolen und der gewünschten Ebene verknüpft werden.

Darüber hinaus bietet der Editor 9 aber auch die Möglichkeit an direkt ein entsprechendes Teileprogramm zu laden und in ASCII-Source-Code darzustellen. Selbstverständlich ist es mit Hilfe des Editors ebenfalls möglich, direkt Anweisungen in ASCII-Source-Code in das Teileprogramm zu schreiben oder den ASCII-Code des Teileprogramms zu verändern. So können z.B. spezielle Anweisungen, sogenannte Strukturieranweisungen als

Schlüsselbegriffe nachträglich, sofern noch nicht bereits vorhanden, in das Teileprogramm geschrieben werden. Eine solche Strukturierungsanweisung enthält neben dem eigentlichen Schlüsselbegriff auch noch Information, wie der durch den Schlüsselbegriff vom Syntaxanalysator erzeugte Schritt bezeichnet wird, nebst optional eine Definition des zugehörigen Symbols sowie optional eine Definition der gewünschten hierarchischen Ebene. Somit können Schrittbezeichnungen, Symbole und Ebenen unabhängig von der Konfigurationsdatei auch direkt in Teileprogramme hinterlegt werden.

Weiterhin bietet der Editor 9 auch die Möglichkeit an, mit Hilfe von Strukturierungsanweisungen, mehrere Schritte zu einem übergeordneten Schritt zusammenzufassen. Diese erlaubt den Aufbau von hierarchischen Schrittstrukturen. Die Schritte können dann in Form einer explodierten Darstellung, bei der alle Schritte angezeigt werden oder implodierten Darstellung bei nur alle Schritte einer z.B. ersten hierarchischen Ebene angezeigt werden, dargestellt werden.

Selbstverständlich ist der Editor auch in der Lage neben der schon oben beschriebenen synchronisierten Schrittdarstellung auch eine synchronisierte Darstellung des Teileprogramms in ASCII-Source-Code darzustellen.

Mit Hilfe einer sogenannten Definitionsdatei, die für den Editor, Informationen über bestimmte Schlüsselbegriffe bzw. Anweisungen enthält, können bestimmte definierte Schlüsselbegriffe und/oder Anweisungen im Editor, insbesondere farblich, hervorgegeben werden. Hierdurch können dem Anwender z.B. herstellerspezifische Anweisungen und Anweisungen nach DIN-Code unterschiedlich dargestellt werden.

Mit Hilfe von sogenannten Zuweisungsparametern können optional auch in der Konfigurationsdatei (8), zu jedem Schlüsselbegriff innerhalb der Konfigurationsdatei (8), mehrere

Schritte zu einem übergeordneten Schritt oder Ebene zusammengefasst werden.

- 5 Ein wesentlicher Vorteil der Erfindung besteht darin, dass unter zu Hilfenahme des Verfahrens, Teileprogramme von Werkzeug- oder Produktionsmaschinen, die in ASCII-Source-Code vorliegen, in Schrittdarstellung angezeigt werden können ohne dass die Teileprogramme verändert werden müssen.

Patentansprüche

1. Verfahren zur Erzeugung und Visualisierung einer aufgabenorientierten Schrittdarstellung (5) von mindestens einem
5 Teileprogramm (1) bei Werkzeug- oder Produktionsmaschinen, dadurch gekennzeichnet, dass das Teileprogramm (1) von einem Syntaxanalysator (7) nach Schlüsselbegriffen durchsucht wird, dass anhand der solchermaßen gefundenen Schlüsselbegriffe eine aufgabenorientierte Schrittdarstellung (5) des Teileprogramms (1) erzeugt und einem Be-
10 diener visualisiert wird.
2. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass als Schlüsselbegriffe fest definierte Anweisungen, insbesondere Synchronisationsanweisungen
15 und/oder Anwenderzyklusanweisungen, bzw. Syntaxteile der Anweisungen verwendet werden.
3. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass die Schlüsselbegriffe sowie deren Zuordnung zu den jeweiligen Schritten in einer vom Anwender veränderbaren Konfigurationsdatei (8) abgelegt sind und vom Syntaxanalysator eingelesen
20 werden.
4. Verfahren nach Anspruch 3, dadurch gekennzeichnet, dass die jeweilige Bezeichnung der Schritte in der Konfigurationsdatei (8) abgelegt ist.
5. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass die Visualisierung mittels eines Editors (9) durchgeführt wird.
30
6. Verfahren nach Anspruch 5, dadurch gekennzeichnet, dass durch den Editor (9) mindestens zwei
35 Teileprogramme gleichzeitig in Schritt- und/oder ASCII-Source-Code Darstellung visualisiert werden.

7. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass die Schrittdarstellung als normierte und/oder synchronisierte Schrittdarstellung (5,10) visualisiert wird.

5

8. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass die einzelnen durchgeführten Aufgaben in der Schrittdarstellung (5) für jeden Schritt durch schrittspezifische Symbole (4a,4b,4c) graphisch dargestellt werden.

10

9. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass mit Hilfe von Strukturierungsanweisungen im Teileprogramm (1)

15 und/oder mit Hilfe eines Zuweisungsparameters in der Konfigurationsdatei (8), mehrere Schritte zu einem übergeordneten Schritt oder einer hierarchischen Ebene zusammengefasst werden.

20 10. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass mit Hilfe von Strukturierungsanweisungen im Teileprogramm (1), Schrittbezeichnungen, Symbole (4a,4b,4c) und/oder hierarchische Ebenen unabhängig von der Konfigurationsdatei (8) auch
5 direkt im Teileprogramm (1) hinterlegt werden.

11. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass mit Hilfe einer Definitionsdatei eine Hervorhebung von definierten An-
30 weisungen und/oder Schlüsselbegriffen auf einem Editor visualisiert wird.

Zusammenfassung

Verfahren zur Erzeugung und Visualisierung einer aufgabenorientierten Schrittdarstellung

5

Die Erfindung betrifft ein Verfahren zur Erzeugung und Visualisierung einer aufgabenorientierten Schrittdarstellung (5) von mindestens einem Teileprogramm (1) bei Werkzeug- oder Produktionsmaschinen, wobei das Teileprogramm (1) von einem Syntaxanalysator (7) nach Schlüsselbegriffen durchsucht wird, dass anhand der solchermaßen gefundenen Schlüsselbegriff eine aufgabenorientierte Schrittdarstellung (5) des Teileprogramms (1) erzeugt und einem Bediener visualisiert wird. Das Verfahren ermöglicht somit eine einfache und kostengünstige Möglichkeit Teileprogramme von Werkzeug- oder Produktionsmaschinen, die in ASCII-Source-Code vorliegen, als Schrittdarstellung (5) anzeigen zu lassen ohne dass die Teileprogramme verändert werden müssen.

20 FIG 2

FIG 1

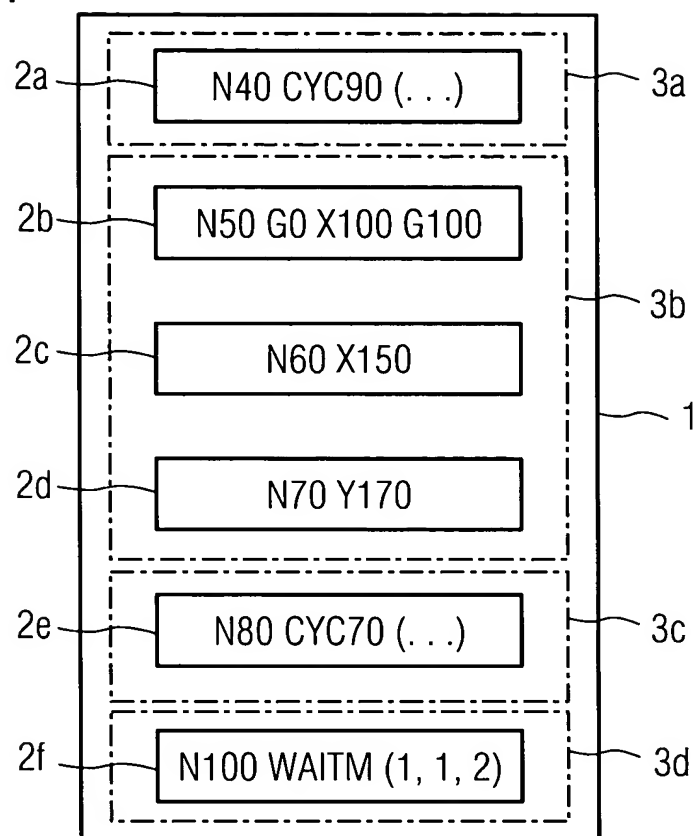


FIG 2

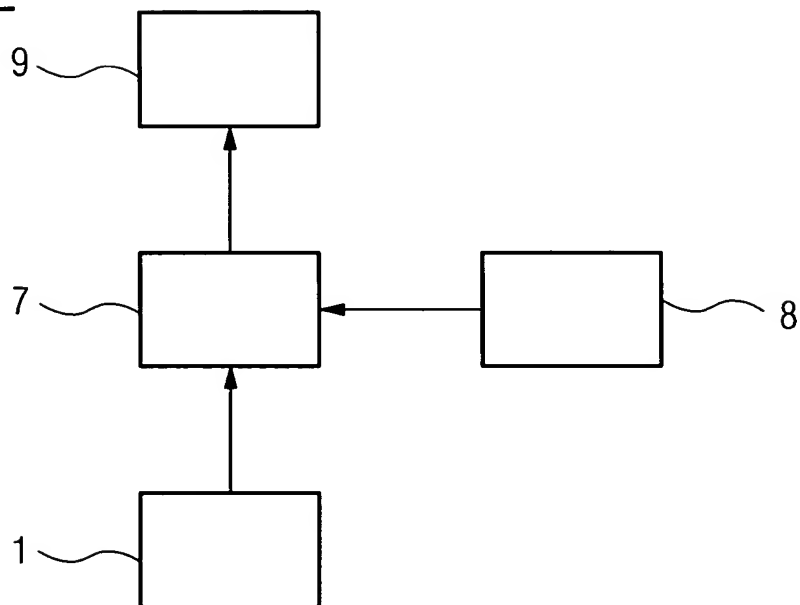


FIG 3

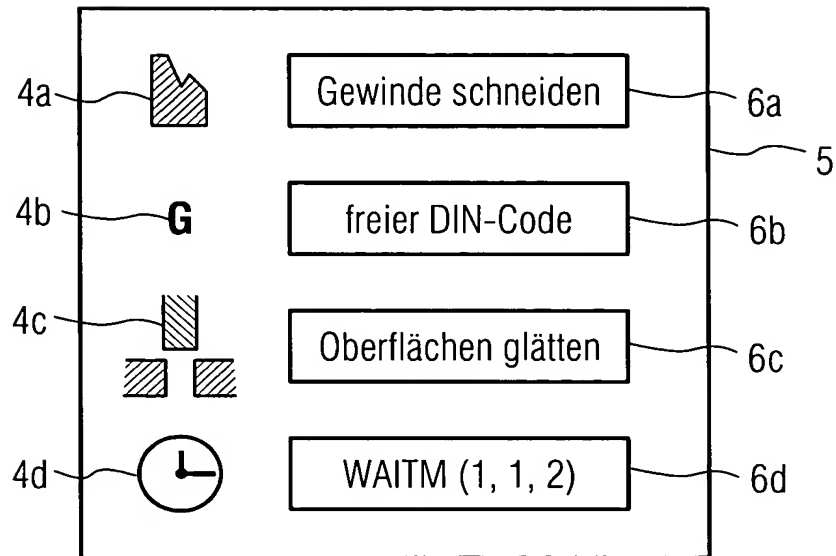


FIG 4

